BAB III

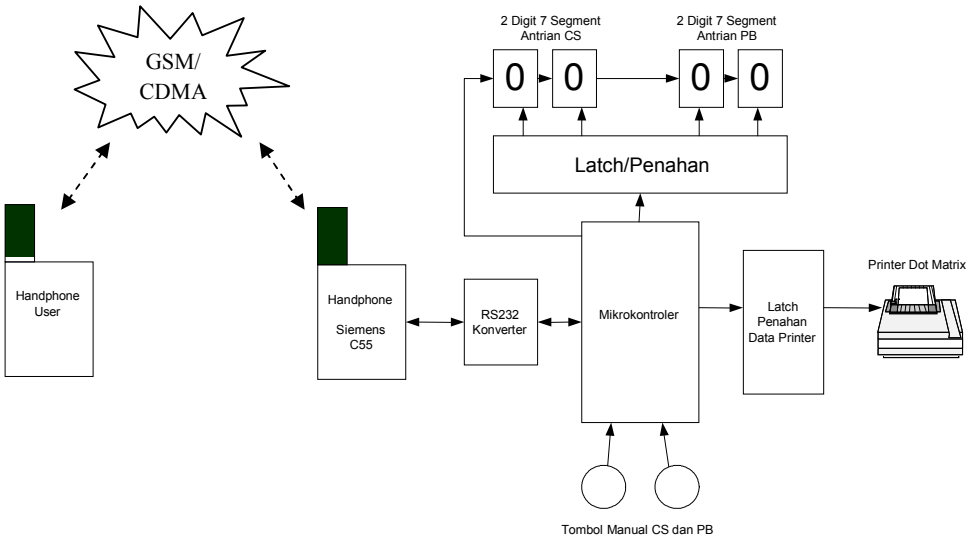
PERANCANGAN DAN PEMODELAN SISTEM

Pada bab ini akan dibahas sistem perencanaan dan pembuatan perangkat keras dan perangkat lunak, yang meliputi.

1. Proses kerja sistem
2. Perancangan dan pembuatan perangkat keras (*hardware*)
3. Perancangan dan pembuatan perangkat lunak (*software*)

3.1. Proses Kerja Sistem

Dalam pembuatan sistem ini maka yang pertama kali dilakukan adalah dengan mengetahui proses kerja sistem yang akan dibuat. Berikut ini adalah blok diagram yang akan dibuat.



Gambar 3.1 *Blok Diagram Keseluruhan*

Proses kerja sistem yang akan dibuat yaitu.

1. Pemesanan Menggunakan SMS

   - User atau pelanggan mengirimkan pesan SMS ke nomor service sistem pelayanan tersebut dengan isi pesan “CS “ untuk mendapatkan antrian pelayanan *customer service* dan pesan “PB” untuk mendapatkan antrian pelayanan pembayaran.
   - Sistem akan memeriksa apakah pesan tersebut benar atau salah. Jika pesan tersebut tidak benar (bukan CS atau PB) maka sistem akan menolaknya dan proses dari sistem tidak akan dilanjutkan. Tetapi jika pesan tersebut benar (CS atau PB) maka sistem akan melanjutkan prosesnya dan mencari nomor antrian yang terbaru.
   - Setelah itu sistem akan mengirimkan kembali pesan SMS kepada pengirim tadi berupa nomor antriannya (misalkan nomor antrian “10”) diikuti kode pelayanan (misalkan “A”).
   - Nomor antrian yang diterima oleh pemesan melelui SMS tadi digunakan sebagai bukti antrian kepada loket-loket antrian.

2. Pemesanan Menggunakan Tombol Manual

   - User atau pelanggan datang secara langsung ke tempat pemesanan nomor antrian.
   - User menekan tombol “CS” untuk mendapatkan nomor antrian pelayanan *customer service* dan menekan tombol “PB” untuk mendapatkan antrian pelayanan pembayaran.

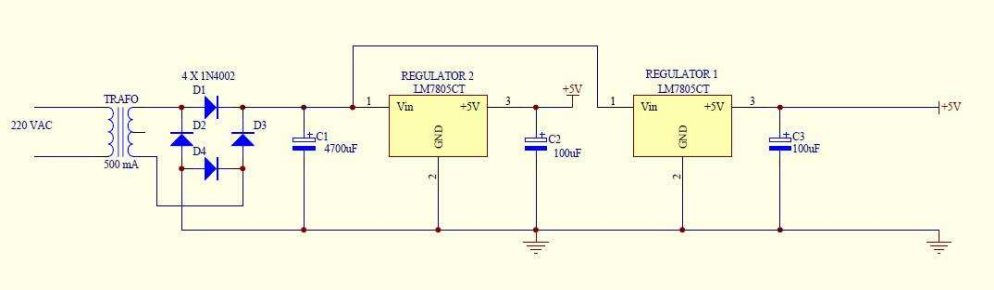
- Setelah user atau pelanggan menekan salah satu tombol tersebut maka mikrokontroler akan memproses permintaan nomor antrian dan di lanjutkan dengan proses pencetakan pada printer dotmatrik.
- Nomor antrian yang diterima berupa nomor antriannya dan diikuti kode pelayanan.

3.2. Perancangan dan Pembuatan Perangkat Keras (*Hardware*)

3.2.1 Catu Daya

Bagian catu daya merupakan bagian terpenting bagi semua rangkaian. Catu daya berfungsi untuk memberikan suplai tegangan, khususnya ke IC mikrokontroler AT89C52 dan modul rangkaian lainya bekerja pada level tegangan 5 Volt DC. Regulator DC tersebut menggunakan rangkaian W*heatstone* yang terdiri dari empat buah dioda sebagai penyearah, dan bekerja dalam mode penyearah gelombang penuh dengan sistem jembatan. *Transformator* yang digunakan adalah jenis CT (*Center Tap*), sedangkan dioda yang digunakan adalah *bridge* yang berfungsi sebagai penyearah gelombang penuh dari AC ke DC. *Transformator* jenis CT dipilih karena dapat difungsikan juga untuk jenis nol. Kapasitor elektrolit digunakan untuk menghilangkan *ripple* akibat penyearahan yang belum sempurna, dengan adanya muatan dari elko maka *ripple* bisa di tutupi. Tegangan keluaran penyearah belum bisa stabil pada satu titik yang diinginkan, misalnya pada 5 Volt DC. Untuk mengatasi hal ini maka dibuatlah catu daya yang dilengkapi dengan IC *regulator*. IC *regulator* yang digunakan adalah LM 7805, dimana IC ini akan meregulasi tegangan mendekati 5 Volt DC sesuai kebutuhan rangkaian. Tegangan

keluaran dari IC ini sudah mendekati tegangan yang diinginkan ±5 Volt DC. Skema rangkaian catu daya diperlihatkan pada ***Gambar 3.2***.

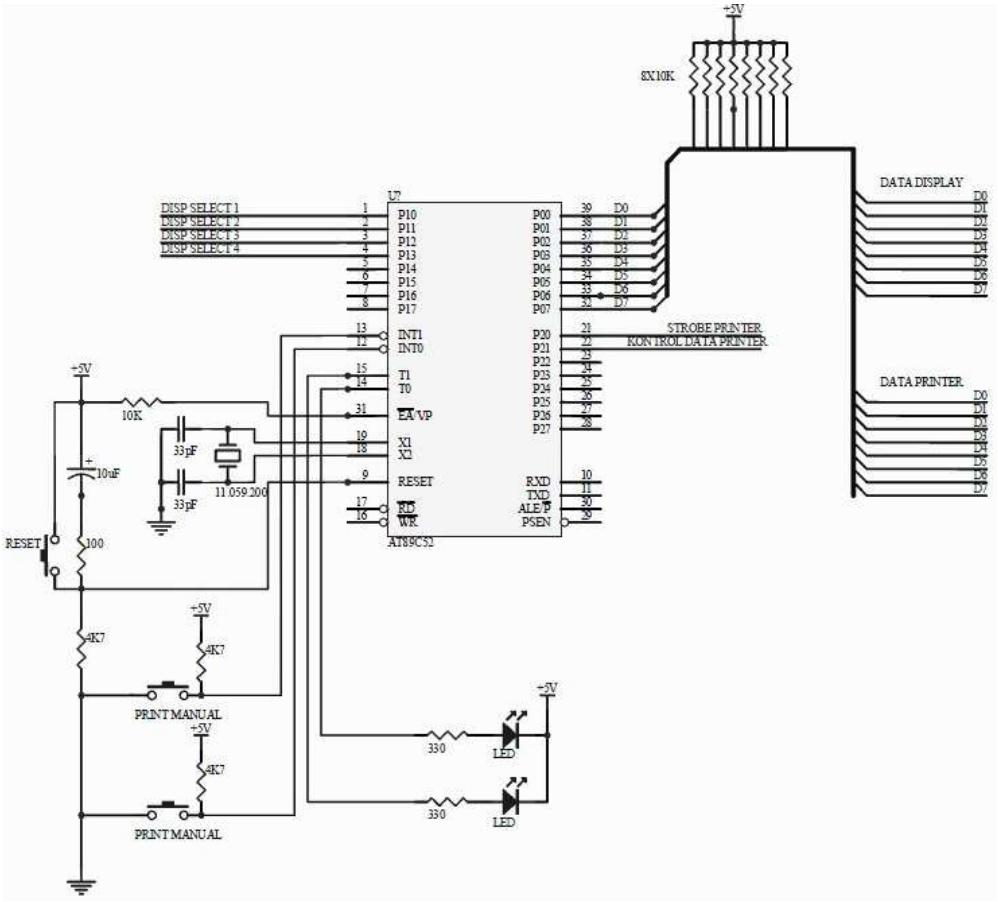


Gambar 3.2 *Rangkaian Catu Daya*

3.2.2 Mikrokontroler

Sistem utama dari perancangan alat ini terletak pada rangkaian mikrokontroler, yang merupakan rangkaian aktif. Sedangkan *handphone* adalah rangkaian pasif. Rangkaian mikrokontroler ini berguna untuk memberikan sinyal reset pada saat alat dioperasikan sehingga program pada mikrokontroler dapat dieksekusi mulai alamat 0000H, selain itu alat ini dapat juga direset secara manual.

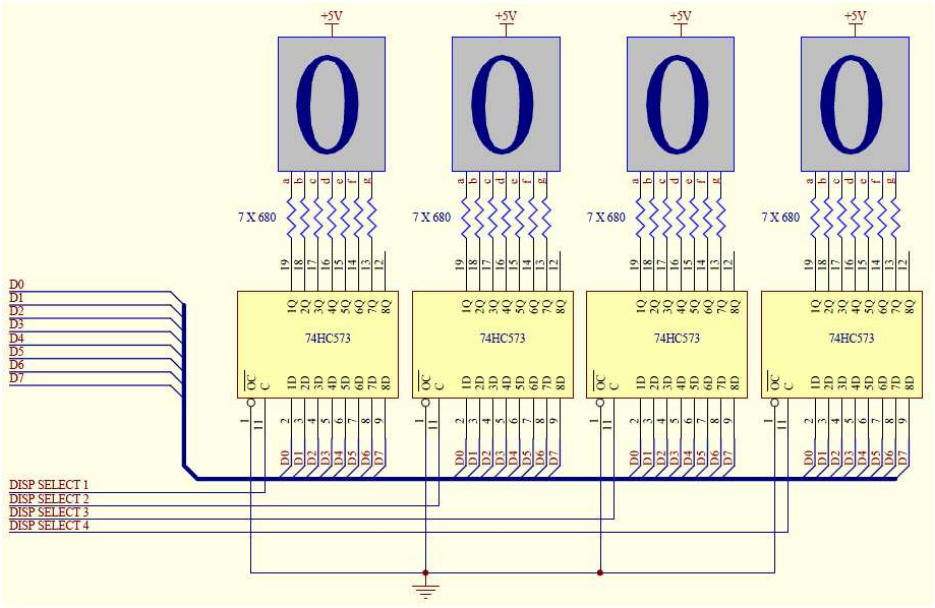
Agar mikrokontroler dapat bekerja, mikrokontroler memerlukan *sinyal clock* yang dibangkitkan oleh sebuah *x-tal* sebesar 11,0529MHz, yang terhubung dengan kapasitor keramik 30pF yang dihubungkan dengan pin 12 dan 13 mikrokontroler. Untuk rangkaian reset, diperlukan kapasitor (C) 10µF dan resistor (R) 4K7 Ω yang dihubungkan dengan pin 31 dan pin 9 mikrokontroler. Sedangkan Vcc dan *Ground* dihubungkan pada pin 10 dan 11 mikrokontroler.



Gambar 3.3 *Rangkaian Mikrokontroler*

3.2.3 Rangkaian *Display Seven Segment*

Untuk menampilkan lebih dari satu peraga *seven segment* yang disusun secara paralel digunakan teknik scaning dimana *seven segment* dinyalakan secara bergantian. Skema rangkaian *seven segment* di perlihatkan pada ***Gambar 3.4.***



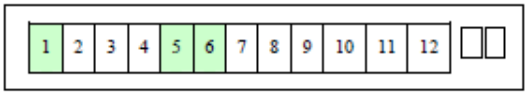
Gambar 3.4 *Skema Rangkaian Seven Segment*

Seven segment tersusun atas LED-LED dengan tegangan dibawah 5 volt maka dipasang resistor 680Ω untuk membatasi arus dengan tegangan di atas LED *seven segment*. Scaning diurutkan sesuai dengan data yang akan ditampilkan. BCD 74HC573 adalah pengkode biner ke desimal untuk data-data angka yang akan ditampilkan ke *seven segment* dan mengirimkan logika low (0) ke kutub anoda. Pin 2, 6, 1, dan 7 (DCBA) pada IC 74HC573 dihubungkan ke empat bit LBS mikrokontroler satu pada port 0 dan port 1, dalam hal ini port yang digunakan adalah port 0.0, port 0.1, port 0.3 dan port 1.0, port 1.1, port 1.2, port 1.3.

Data hexa hanya dikirimkan ke IC 74HC573 yang akan dikonversi menjadi bilangan desimal kemudian ditampilkan di peraga *seven segment*.

3.2.4 *Interface Handphone* ke Mikrokontroler

Koneksi dengan *handphone* Siemens pada dasarnya digunakan untuk dapat berkomunikasi dengan *handphone*, sehingga diperlukan sebuah level konverter. Konektor *handphone* Siemens C55 dapat dilihat pada ***Gambar 3.5***. Tidak semua pin out terhubung ke mikrokontroler, tetapi hanya pin nomor 1 (*ground*), 5 (*Tx/data out*) dan 6 (*Rx/data in*).



Gambar 3.5 *Konektor Handphone C55*

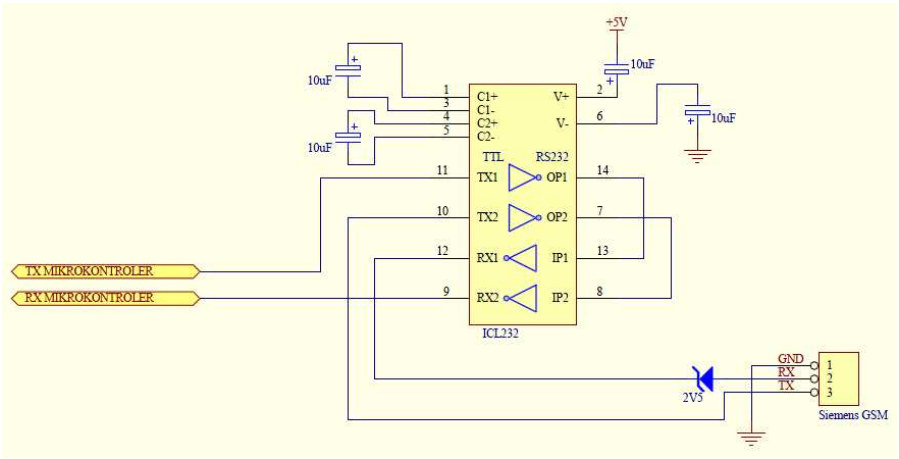
Untuk penggunaan dari masing-masing pin dari konektor pada ***Gambar 3.5*** dapat dilihat pada ***Tabel 3.1.***

Tabel 3.1 *Pinout Konektor C55*

PIN	NAMA	FUNGSI	IN/OUT
1	GND	Ground	-
2	SELF SERVICE	Recognition/control battery charger	In/Out
3	LOAD	Charging Voltage	In
4	BATTERY	Baterry	Out
5	DATA OUT	Data sent	Out
6	DATA IN	Data received	In
7	Z_CLK	Recognition/control accessories	-
8	Z_DATA	Recognition/control accessories	-
9	MICG	Ground for microphone	In
10	MIC	Microphone input	-
11	AUD	Loudspeaker	Out
12	AUDG	Ground for eksternal speaker	-

3.2.5 Koneksi RS-232

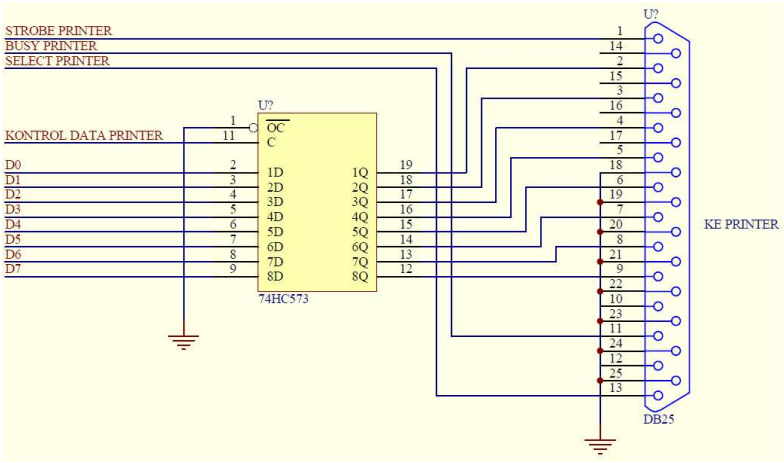
Untuk dapat terhubung secara *hardware* antara *handphone* dengan mikrokontroler perlu adanya suatu *converter* untuk menyesuaikan level tegangan TTL antar dua *hardware* tersebut, yaitu dengan menggunakan RS232. Dimana level tegangan pada mikrontroler yang digunakan adalah 0 Volt dan +5 Volt, sedangkan level tegangan pada *handphone* adalah 0 Volt dan 3 Volt. Komunikasi MAX232 dengan mikrokontroler yaitu dengan menghubungkan kaki Rx mikrokontroler (pin 14) dengan TX1 MAX232 (pin 11), dan Tx mikrokontroler (pin 15) dengan kaki RX2 MAX232 (pin 9).



Gambar 3.6 *Rangkaian MAX232*

3.2.6 *Interface* Mikrokontroler ke Printer

Rangkaian Interface Printer berfungsi untuk mencetak nomor antrian pada saat tombol manual digunakan, rangkain ini menggunakan IC 74HC573 yang berfungsi sebagai penahan data atau konversi dari mikrokontroler ke media percetakan dalam hal ini printer dotmatrix. Adapun rangkaiannya dapat dilihat pada ***Gambar 3.7***.



Gambar 3.7 *Rangkaian Interface Penahan Data Printer*

Tabel 3.2 *Resume Keseluruhan Komponen Hardware*

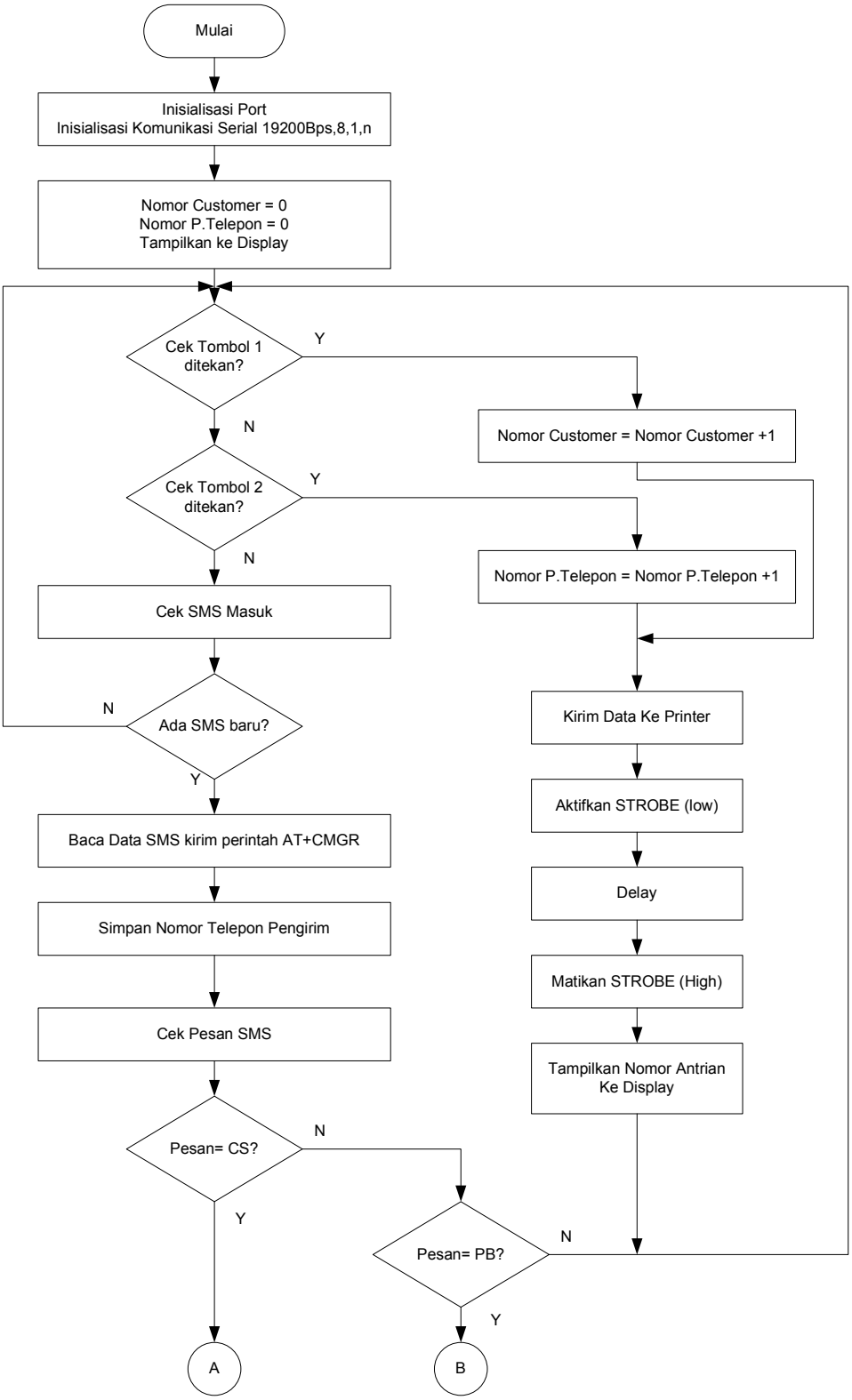
No.	Nama Komponen	Fungsi
1	Mikrokontroler ATMEL AT89C52	Otak dari rangkaian, mengintegrasikan seuruh proses yang ada.
2	ISP (*In System Programmer*) dengan DB-25	Untuk pemograman mikrokontroler
3	PCB	Penghubung antar komponen
4	*Handphone* dan Kabel data	Mengirimkan SMS di mokrokontroler, mengirim dan menerima SMS di SMS server
5	Catu daya	Memasok daya yang dibutuhkan kerangkaian mikrokontroler
6	PC/notebook dengan program adesign Explorer DXP, Hyperterminal/Minicom	Melakukan pemograman AT89C52.
7	Multimeter	Untuk mengukur tegangan
8	MAX323, LM7805, kristal, resistor, kapasitor, transistor, dioda, DB9, LED, Socket 8/16/40 pin, header power	Komponen rangkaian mikrokontroler
9	Solder, timah, kabel	Komponen dasar untuk merakit rangkaian

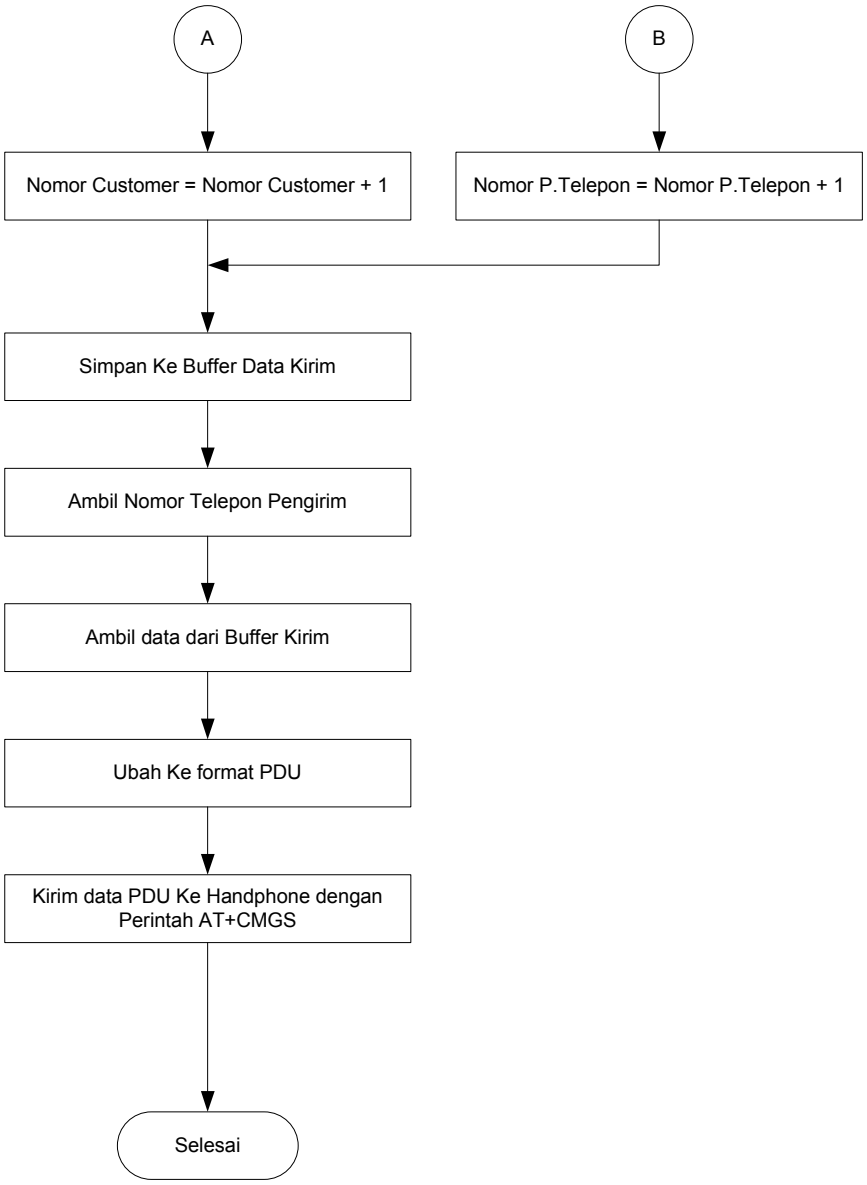
3.3. Perancangan Perangkat Lunak (*Software*)

Software diperlukan sebagai protokol antara *handphone* dengan mikrokontroler juga komunikasi antara mikrokontroler. *Software* juga digunakan untuk mengolah data input serta mengendalikan output mikrokontroler dalam proses pengiriman SMS dari *handphone* menggunakan protokol PDU. Artinya mikrokontroler harus mengikuti protokol PDU pada *handphone* yang digunakan, dalam hal ini Siemens C55. Sistem komunikasi antara *handphone* dengan mikrokontroler terjadi satu arah yaitu transmit (*deliver*). Pengiriman pesan atau SMS *Submit* dari *handphone* server ke *handphone* user menggunakan jalur serial (*serial port*) dari mikrokontroler. Mikrokontroler menyesuaikan *baud rate handphone*, yaitu 19200 bps (*bit per second*).

3.3.1 Program Utama

Main program atau program utama menunjukkan proses mikrokontroler secara global. Alur programnya dapat dilihat pada ***Gambar 3.8***.


Mulai
Inisialisasi Port
Inisialisasi Komunikasi Serial 19200Bps,8,1,n
Nomor Customer = 0
Nomor P.Telepon = 0
Tampilkan ke Display
Cek Tombol 1 ditekan?
Y
N
Nomor Customer = Nomor Customer +1
Cek Tombol 2 ditekan?
Y
N
Nomor P.Telepon = Nomor P.Telepon +1
Cek SMS Masuk
Ada SMS baru?
N
Y
Kirim Data Ke Printer
Aktifkan STROBE (low)
Delay
Matikan STROBE (High)
Tampilkan Nomor Antrian Ke Display
Baca Data SMS kirim perintah AT+CMGR
Simpan Nomor Telepon Pengirim
Cek Pesan SMS
Pesan= CS?
N
Y
Pesan= PB?
N
Y
A
B



Gambar 3.8 *Flowchart Utama*

Pertama-tama setelah start program diikuti proses inisialisasi dari port yang digunakan untuk komunikasi serial dengan *boutrate* 19200 Bps dan dilanjutkan dengan memeriksa nomor *customer center*, selanjutnya sistem memeriksa apakah ada SMS masuk atau tidak jika terdapat SMS masuk maka sistem akan membaca data dari SMS tersebut lalu nomor pengirim tersebut akan disimpan di dalam

buffer untuk beberapa saat. Selanjutnya data dari SMS tersebut dibaca apakah pesan tersebut berupa CS ataukah PB, jika pesan tersebut berupa kode CS atau PB maka sistem akan mencari nomor antrian yang sekarang ditambah satu, jika pesan tersebut bukan berupa kode CS dan PB maka sistem akan membuangnya. Selanjutnya sistem akan mencari nomor telepon yang terdapat dalam *buffer* di lanjutkan dengan mengubah ke dalam format PDU dan diteruskan dengan mengirimkan kembali ke nomor pemesan nomor tersebut.

3.3.2 Program Pengkodean Isi SMS Dalam Format Biner

Setelah user mengirimkan SMS yang berisi "CS" atau "PB" maka dilakukan proses pengkodean isi SMS. Pada proses pengkodean isi SMS ada dua langkah yang harus kita lakukan untuk mengkonversikan isi SMS, yaitu.

1. Langkah pertama : mengubahnya menjadi kode 7 bit.
2. Langkah kedua : mengubah kode 7 bit menjadi 8 bit, yang diwakili oleh pasangan heksa.

Semua kode ASCII 8 bit dari 18 angka dan huruf yang ditekan dijadikan 7 bit. Dengan cara di-ANDkan dengan 01111111B. Kemudian data tersebut dimasukkan ke pointer 1 s/d pointer 18 dan data tersebut digeser ke kanan satu kali. *Carry* hasil geseran pointer 2 ditambahkan ke data pointer 1. Pointer tersebut tersebut dipindahkan dari pointer 1 menjadi pointer 2 dan berulang-ulang sampai dengan pointer 18. kemudian cek apakah sudah sampai pointer 18. Jika tidak, maka akan kembali ke pengeseran data 1 kali. Jika ya, maka pointer 2 jadi pointer 1 dan pointer 3 jadi pointer 2 dan seterusnya sampai pionter 18. lalu cek apakah pointer 3 sama dengan pointer 18. Jika tidak, maka looping ke geser satu kali dst.

Jika ya, maka pengkodean sudah selesai dan disimpan di SRAM. Dan nanti akan diambil ketika menjalankan program pengiriman SMS.

3.3.3 Program Pengiriman SMS

Setelah proses pengkodean isi SMS maka dilakukan proses pengiriman SMS. Pada proses pengiriman SMS ini menggunakan *handphone* Siemens C55, dan diperlukan penyesuaian protokol dengan baudrate 19200bps sebagai inialisasi program.

Untuk mengirimkan SMS diperlukan instruksi *AT Command* yaitu dengan mengirimkan perintah AT+CMGS=19, maka isi data pesan tersebut maksimal 24 karakter dimulai dari tipe SMS sampai dengan isi SMS. Kemudian kirim enter dengan code heksa 0DH dan 0AH. Lalu ponsel akan merespon perintah dengan simbol “>” atau dalam heksa $3E. Setelah itu tahap kedua yaitu pengiriman header yang terdiri dari nomor SMS-center, Tipe SMS, Nomor Referensi SMS, Nomor Ponsel Penerima, Bentuk SMS, Skema *Encoding Data* I/O, Jangka Waktu Sebelum SMS *Expired*, panjang isi SMS. Lalu dilakukan pemrosesan data pesan dalam kode biner untuk dirubah menjadi kode heksa, kemudian dikirim.